Automated Analysis of Dynamic Single Molecule Experiments

Undergraduate Thesis

Presented in partial fulfillment of the requirements for graduation with *research distinction* in Physics in the undergraduate colleges of The Ohio State University

by

James London

The Ohio State University

April 2018

Project Advisor: Professor Richard Fishel, Department of Physics and Department of Cancer Biology and Genetics

Abstract

Single molecule total internal reflection fluorescence studies using co-localization and fluorescence resonance energy transfer have become valuable tools to study biological phenomenon. However, no comprehensive programs are publicly available to quickly analyze the videos generated from these studies. Here we present several methods to automatically analyze these videos and a user oriented graphical interface to allow researchers to easily run these analyses. For each method the algorithms used are explained and their performance is evaluated. Finally, we explain the design of the user interface and its operation.

Acknowledgements

I would like to thank my advisor, Professor Richard Fishel. His encouragement, enthusiasm, and support for me and the project have been vital. I would like to thank Jiaquan Liu for encouraging me to first pursue this project and for his advice as the project came together. I would like to thank Jeungphill Hanne for teaching me about single molecule experiments, pushing me to work hard, and supporting me when I was down. I would like to thank Gayan Senavirathne for helping me talk through the more difficult parts of the project and for inspiring me. I would like to thank Professor Kristine Yoder for all her help with my summer fellowship. I would like to thank all the members of Fishel Lab and Yoder Lab who have taught, encouraged, and supported me. I would like to thank Professor Ralf Bundschuh for sitting on my thesis defense committee. And finally, I would like to thank my parents for making me who I am and for first encouraging me to program.

This research was supported by the Office of Undergraduate Research & Creative Inquiry Summer Research Fellowship.

Table of Contents

Introduction

Background

Single molecule imaging techniques allow researchers to examine biology on the scale of proteins and DNA. Molecules can be directly observed one at a time, allowing researchers to observe phenomena that vary between complexes and are only visible on the molecular scale.

One such method is single-molecule total internal reflection fluorescence (smTIRF). It uses a TIRF field, produced by a laser reflecting off a slide buffer interface beyond the critical angle to illuminate a small section of a slide with an evanescent field. This allows biologically relevant concentrations to be used while limiting the field of view so that individual molecules can be distinguished. Observed molecules are labeled with a fluorescent dye, and several different dyes can be used to allow different types of molecules to be observed. The experiments are recorded using an inverted microscope and each range of fluorescent wavelengths are separated by filters. The separated wavelengths are projected onto separate sections of a camera chip. This results in a recording where each color channel appears in a separate section of the video.

Two common types of experiments used in smTIRF are co-localization studies and single molecule fluorescence resonance energy transfer (smFRET). In co-localization studies, both ends of a DNA are tethered to the slide. Labeled molecules can then interact with the DNA with the fluorophores directly activated by the lasers. The video from these studies are analyzed by generating a kymograph along the DNA. The labeled complexes can then be characterized through their interaction with the DNA and with each other on the DNA. Currently, these videos are analyzed by researchers watching the recordings and extracting data from their observations or using custom scripts that are not publicly available and often only perform a small portion of the needed analysis.

In FRET, two or more fluorophores are used such that one fluorophore, the donor, can transfer energy to another, the acceptor, through a distance-dependent dipole-dipole coupling[1]. This interaction results in an anti-correlated donor and acceptor fluorescence as the two come close together. In smFRET, these experiments are typically performed by tethering one fluorescently labeled complex to the slide while another interacts with it[2]. Traces are generated by analyzing the intensity over time at the location of each fixed molecule. From these traces

researchers can develop an understanding of the biological interactions taking place. Currently, there are several publicly available programs to analyze smFRET, but they use unautomated methods to manually combine the separated channels using insufficient rigid transformations[2,3].

Overview

Here we present a set of computational methods to automate the analysis of smTIRF co-localization studies, smFRET studies, and a complete graphical user interface (GUI) to allow researchers to perform these analyses. The analysis aligns the color channels, detects particles, detects DNA, analyzes kymographs, and analyzes FRET. The GUI will be made publicly available shortly, as it is currently being tested before release. The program and methods discussed were all developed in Matlab 2016b and are available at https://github.com/7jameslondon/Albus[4]. Both a pre-compiled standalone version of the GUI and the original Matlab functions are provided.

Detection and Localization of Particles

Overview

The detection of fluorescent particles, such as beads and molecules, is used in three aspects of the analysis: to find the beads for channel registration, to track the molecules to find the DNA location, and to find the location of the FRET molecules. The same analysis is used for each of these applications.

Methods

First, the particles must be identified. Afterword, their exact position can be determined. To find the particles in an image, the image is segmented into separate regions ideally with one region for each particle. Several methods were reviewed from the literature[5], and a local maximum method was chosen based on the performance evaluations in reference [6]. The local maximum is found by first convolving the image with a circular Gaussian kernel. The user provides the standard deviation of the Gaussian kernel, which is easily estimated by adjusting the size until the particles are uniform but not merged as shown in Figure 1b. The image is then broken into sections around each local maximum. Last, the user can select which regions will be accepted as particles by choosing a range of acceptable mean intensities.

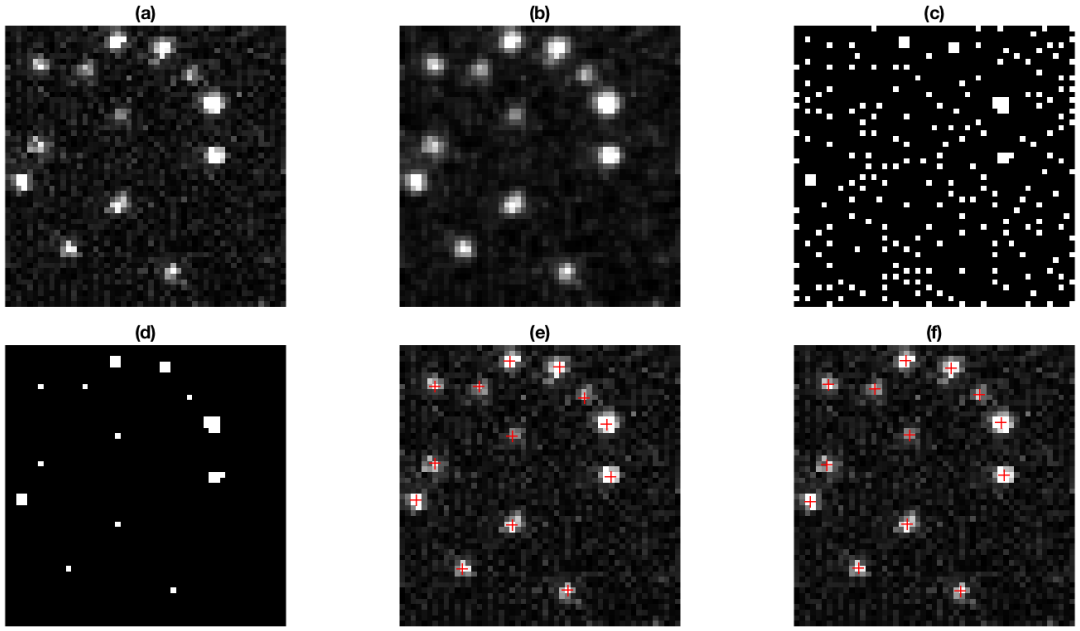

Figure 1 Steps to segment and detect images. (a) Original image of fluorescent beads. (b) Image after convolving with a Gaussian kernel of standard deviation 0.5. (c) Binary image, with only local maximums shown. (d) Binary image, with only local maximums with a mean intensity above 0.5 shown. (e) Original image with overlaid

estimates of particle positions using the intensity weighted mean of the pixel positions. (f) Original image with overlaid sub-pixel particle positions by linear least-squares circular 2D-Gaussian fitting.

Next, the particles' positions must be determined. This can be done by several methods which vary in accuracy and calculation time. Ideally, the particles would be fit to a point spread function specific to the experimental setup[7]. However, in practice, a 2D-Gaussian is just as accurate. Fitting a 2D-Gussian can be quite computationally intensive. In several commonly used scripts, the position of the peak value is used[8]. However, with advances in computational speed, parallel processing, and a good starting estimate of the parameters, 2D-Gusiasian fitting can be very fast[9]. The position can be estimated using the intensity weighted mean of the particle pixels. The height can be estimated using the intensity at this position and the standard deviation using the average radius of the region. To perform the fitting, linear least-squares regression is used. The center of this Gaussian will then provide the sub-pixel position of the particle.

Performance

To test the accuracy of this method, 1000 50x50 pixel artificial images were generated. First, a set of circular 2D-Gaussian functions were created from a set of random positions, widths, and heights. Then, the Gaussians were randomly sampled to simulate the stochastic emission from a fluorophore. Last, Gaussian noise was added to simulate the background noise observed in experiments from a combination of electronic noise and the fluorescent background[7].

The intensity weighted mean method and the least-squares fitting method were run on the images and the positions from each localization method were compared to the original positions. Because the segmentation process could generate more or fewer centers than there really were, the minimum distance to any center was used. The average minimum distance was 0.361 pixels for just the intensity weighted mean method and the average minimum distance was 0.348 pixels for the linear least-squares fit. Clearly, this gain is rather marginal with an absolute difference of 0.013 pixels and a relative difference of only 3.74%.

The speed of these methods was also tested using the same data set on several computers including two personal laptops and two desktops. All computers had 4 cores and were running no other programs when tested. Only the least-squares fitting portion of the fitting method was run in parallel using one thread per core while the segmentation and intensity weighted mean were run linearly. On average, the intensity weighted mean method took 7.00 ms of wall time to compute per image and the fitting method took 223.041 ms.

This larger computation time for such a small gain would be cumbersome when tracking particles throughout an entire typical 512x512 pixel video. However, when analyzing a single image, the total time for detection using the least-squares fitting method rarely exceeds 1 minute. Thus, for computing the particle positions for the channel registration and FRET positions, where only one image is needed, the fitting method is used. As a result, the fitting method significantly enhances the channel registration as discussed in the next section.

Channel Registration

Overview

In experiments where multiple types of fluorophores are used, the different wavelengths of emitted light are separated by filters and projected onto separate sections of a EMCCD camera chip. These channels must be recombined to analyze the relative positions of the molecules. Due to the nature of the filtering and projections, the channels are not simply translations of each other along the chip. Nor are they affine transformations but are instead each locally deformed by the lenses and filters they pass through, such as from a "fish-eye" effect. These differences can be accurately compensated by determining the thin-plate spline (TPS) that transforms the coordinates of one channel to the next[8,10].

Typically to perform this alignment, a movie is first recorded using multi-fluorescent beads which appear in all channels[2]. The movie is then separated into each section of the channels. The channels can then be aligned using the positions of the beads in each channel such that they will all overlap after the alignment.

Methods

The beads in each channel can then be localized using the technique described in the previous section. Next, the beads must be matched up with each other. Then, the TPS transformation of each channel can be calculated and applied to transform all the channels to the same coordinate space.

The first issue is to match up the beads. Differences in the quality of the video from each channel and inaccuracies in the detection algorithm can lead to different sets of beads being found in each channel. Thus, only a subset of the beads found in each channel will be the same beads. This subset of beads must be found in each channel and then ordered to match with their corresponding beads in all other channels. To do this efficiently, three random bead positions are selected from the second channel. Then, their corresponding neighbors in the first channel are found within a given radius[*]. Each combination, of three neighboring beads is evaluated. For each combination the affine transformation from the position of the three second channel beads

[*] This radius can be automatically determined by using the median distance of each bead to the next in the first channel. This should result in an optimal set of neighboring beads such that a minimum number of possible neighbors are selected but that most beads have neighbors with their actual corresponding bead. However, in the GUI the user is given the option of extending this radius if the analysis does not find a suitable transformation.

to the three in the first channel is found. This transformation is used to transform all the beads in the second channel. Then, the sum of the distance between each bead in the transformed second channel to the nearest neighbor in the first channel is calculated. If this is less than a given tolerance the transformed beads are selected. If not, a different combination of neighboring beads in the first channel is tried. When they are exhausted, another set of three random beads from the second channel is selected and the iteration of neighbors is restarted. This process is repeated for each channel with the first channel. After this process, the channels will be approximately overlapped as shown in **Figure 2**b.

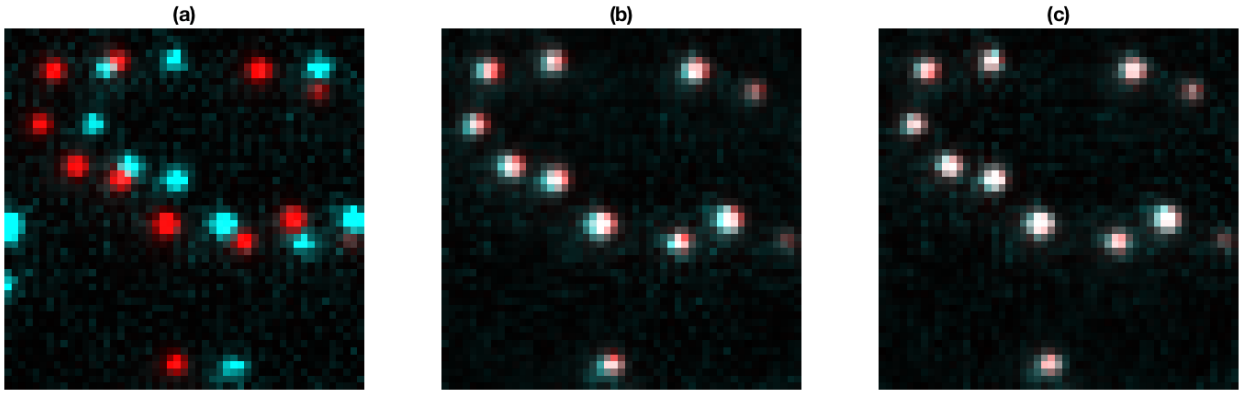

Figure 2 Multi-fluorescent beads in two channels being aligned. (a) The original channels after being separated, colored and imposed on top of each other. (b) The channel after an approximate affine transformation is used to align the blue channel to the red channel coordinate space. (c) The channels after the TPS transformation is used to align the blue channel to the red channel coordinate space.

Next, the approximately overlapped bead positions are used to match the corresponding beads in each channel. To do this matching, the nearest neighbor of each position in the first channel is found in every other channel. If and only if there is exactly one neighbor in every channel within a user provided radius, then the beads are matched. This condition results in a subset of beads in the first channel that have a corresponding bead in every other channel within the given radius. However, this can lead to two or more beads in the first channel "claiming" the same bead in another channel. To simply resolve this conflict all beads that are "claimed" more than once and their corresponding first channel bead are dropped. This process ends with an ordered subset of positions in each channel such that each the index of any element of a channel set is the index of the corresponding positions in all other channels.

These ordered sets of positions are interpolated using a TPS with an optional user provided smoothing parameter. The smoothing parameter relaxes the requirement that a bead's interpolated position be the same as its given position(see reference [11] for further discussion on

TPS smoothing). The interpolation provides a set of TPS weights and control points that are then used to transform the videos of the other channels to correspond with the coordinate space of the first channel. An example of this result is shown in **Figure 2**. Additionally, the surface of the TPS for the data used in figure 2 is shown in figure 3. This surface plot clearly shows the "bending" in the coordinate system. Furthermore, in figure 3c the TPS surface is viewed from above where only the planar transformation is visible. This shows the degree to which the TPS transformation mimics the affine transformation such as through the rotation and scaling of the plane.

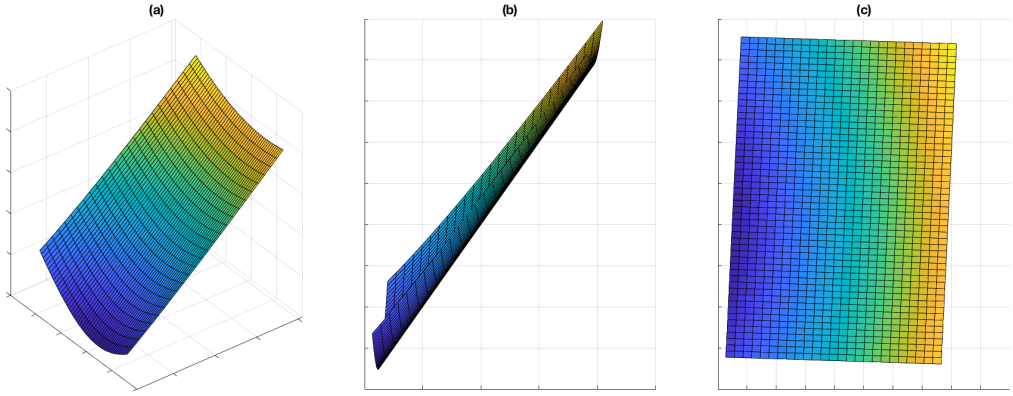

Figure 3 A thin plate spline used to map the beads in figure 2 (a) Isometric perspective. (b) Profile perspective. (c) Arial perspective. In (a) and (b) the "bending" of the surface is visible. In (c) the aspects of the TPS that are also compensated in the affine transformation, such as rotation and scaling, are visible.

Performance

No robust methods are available to evaluate the accuracy of this method. However, artificial data was tested and real data from several experiments was examined by eye for accuracy. The artificial data was created the same way as in the previous section and then transformed with an affine, projective or TPS transformation. The registration analysis was then performed on the data set. No difference between the original Gaussian positions used to generate the artificial data and the positions after the registration was applied to the transformed data was measured beyond the error caused by the detection analysis discussed in the detection and localization section. The real data was examined from 12 experiments. In each the channels were registered as shown in **Figure 2**c. Each of the registered images were examined by eye and appeared to be accurately overlapped.

The entire registration algorithm takes less then 1 second to compute on all computers it was tested on.

DNA Detection and Kymograph Generation

Overview

In co-localization studies, a DNA is typically tethered on each side to the experimental slide. Labeled proteins are injected in the slide and a recording is made. After the experiment, the DNA is stained with dye and a second recording is made, which reveals the location of the DNA. However, in some experiments the DNA is broken when, for instance, a nuclease is being observed. In these cases, the DNA cannot be stained and observed after the experiment. Thus, to determine the position of the DNA, the proteins in the experiment are tracked and 1D diffusions are selected as candidate positions for the DNAs.

In both cases after the position is determined, a kymograph is generated. Kymographs are images where the line of pixels along the DNA are taken at every frame and joined together to create an image that depicts 1D space over time as show in Figure 4. Because these kymographs are used to analyze the actual experimental data, it is critical that they are drawn correctly. For instance, generating a kymograph from slightly above a DNA will skew data to show less activity of proteins on the DNA. In another case, drawing a kymograph diagonally across a DNA will artificially lower the diffusion rate of observed proteins.

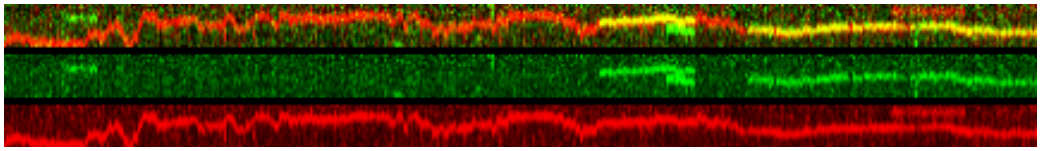

Figure 4 An example of a kymograph with both channels combined (top) and separated (middle and bottom).

Methods

If no stained DNA video is available, the user can choose to analyze the video for 1D diffusions. First, in each frame all the possible particles are identified using the analysis described in the first section. To create tracks a greedy nearest neighbor algorithm is used. This algorithm requires that, for each frame, each particle finds its nearest neighbors within a given radius* in the next frame. For any given particle in a frame, where no nearest neighbor is found in the next frame, then the particle can check the frame after that and so on up until a given

* This radius can be provided by the user or an iterative algorithm will attempt to find an ideal radius. The program checks the first 10 frames using a radius of 1 pixel. The algorithm checks to see how much random 2D diffusion was captured by the tracking. If the most particles are considered stationary or disappear, the radius is increased by 0.5 of a pixel until the criterion is met.

maximum. This requirement allows for short disassociation events and errors in the particle detection algorithm. Enforcing this assumption does not affect the results of the overall analysis, because it is only being used to determine if there is a DNA in the area. If two particles in a given frame choose the same particle in the next frame, the closer particle gets to claim it. Though this algorithm is not robust for tracking in general, it is very fast and suitable for detecting regions with 1D diffusions[6]. The tracks are then combined into a 2D histogram. For each position in a track, it adds a point to the histogram weighted by the total length of the track. Thus, a track with 10 points will add 10 to each of those 10 positions in the histogram. This emphasizes longer tracks which are much more likely to be actual 1D diffusions rather than background noise because they are sustained for longer. This histogram acts like an image of where the DNA should be and can then be analyzed using the same method as the stained DNA.

To extract the DNA positions from the stained DNA, the image is segmented into DNA candidates. Several methods were reviewed from the literature[5] and a clustering method using K-means was selected based on the relevance of the application. The K-means method attempts to separate the image into a given number of K separate groups by minimizing the variance of each group. This is applied using Matlab's *kmeans* function. To determine K, the number of local maximums are counted. Once the groups are separated, they are each characterized by their eccentricity, mean intensity, and major length. The ranges can be automatically approximated using an eccentricity greater than 0.75, a mean intensity greater than the mean intensity of the image, and a range of lengths between ±0.1 standard deviations of the mean length. Alternatively, the user can set the range for each of these values. Once the DNA candidates have been selected, the position of the DNA is determined. The DNA position is treated as a single line segment. Its slope is determined by least-squares fitting a linear curve to the intensity weighted pixel positions. The end points of the line are determined by where the fit curve intersects with the borders of the separated group. This process ends with a set of all the DNA positions.

To generate kymographs, each frame of the video is cubically interpolated along each DNA position with pixels being sampled along the line every 1pixel-width. The interpolation of each frame, for a given DNA, are then sequentially combined.

Performance

The performance of this analysis was evaluated by comparing to the previously published study, reference [12]. In one part of the study two proteins were examined as they diffused along the DNA and a stained DNA video was available. In the study 46 DNAs were identified. Using this analysis without the stained DNA 89 DNAs were identified. Of these 43 were the same as those found in the original study and 3 were missed by this analysis. Furthermore, using the stained DNA to analyze the data, 184 DNAs were found. Of these all of the DNA found by the analysis without the stained DNA and found in the study were found. Thus, 48.6% of the known DNA were found without using the stained DNA video. Whereas the original study only found 25.1% of the known DNAs. This clearly shows a very substantial improvement in data collection using this automated analysis.

This analysis has a wide variety of computational time depending on the number of initial candidate DNAs first identified and the processing speed on the computers used ranging from 10 minuets to 3 hours. Both the k-means algorithm and the kymograph generation can be run in parallel to speed up the respective execution time when possible.

FRET Trace Generation and Analysis

Overview

In smFRET, two or more fluorophores are used such that one fluorophore, the donor, can transfer energy to another, the acceptor, through a distance-dependent dipole-dipole coupling[1]. This interaction results in an anti-correlated donor and acceptor fluorescence as the two come close together. In smFRET, these experiments are typically performed by tethering one fluorescently labeled complex to the slide while another interacts with it[2]. Traces are generated by analyzing the intensity over time at the location of each fixed molecule. From these traces researchers can develop an understanding of the biological interactions taking place. The complexes are typically located by briefly directly exciting the acceptor channel before injecting the donor complex. The positions of all the complexes are visualized from this can then be used to generate the traces. Once the donor and acceptor traces are determined the FRET efficiencies are calculated as described in **Equation 1**[3]. The FRET efficiency over time is then analyzed to show changes in the interaction of the donor and acceptor. These differences in interaction reflect different states the two molecules are in such as conformational changes between two proteins. Often hidden Markov modeling (HMM) is used to find the underlying states in the FRET efficiency trace.

$$E = \frac{I_A}{I_A + I_D}$$

Equation 1 Where E is the FRET efficiency, I_A is the intensity in the acceptor channel of a complex, and I_D is the intensity in the donor channel of a complex.

Methods

First the acceptor positions are determined using the analysis described in the detection and localization section using the frames where the acceptor was directly excited. The positions returned from this analysis are then treated as candidate positions. For each candidate the radius is determined by two times the standard deviation of the Gaussian used to localize the candidate. The candidates whose radius intersects with another candidate are removed. Additionally, candidates whose radius extends past the border of the video frame are removed. This condition removes traces where the particles overlap too much and will thus influence the intensity of their

neighbors. Finally, candidates whose eccentricity* are non-zero are removed. This is to remove candidates that are detected as one complex but are actually two close fluorophores. All of these parameters can be adjusted by the user if the nature of the experiment requires it.

The remaining candidates are then accepted as actual events. Traces are generated by taking the average intensity over the circle formed by the radius over the cubically interpolated image.

To analyze the traces each channel is first fit to a HMM. Several programs have been developed to perform HMM for smFRET efficiencies[3]. One such program is vbFRET which attempts to model a trace with a variety of HMMs choosing using a maximum likelihood approach to choose the model with the fewest possible states and the best fit[13]. The authors of vbFRET provide an open-access version of the program which is integrated into the analysis presented here. The lowest intensity value of the HMM for each channel is then used as the background intensity. Using the background subtracted traces the FRET efficiencies can then be calculated using equation 1. The efficiency is finally fit with a HMM using vbFRET and the dwell times in each state histogrammed. This provides the user with the relevant data necessary to understand the relevant characteristics of their experiment.

Performance

The performance of this analysis was easily compared to previous thresholding analyses, such as that used in reference 14, because it uses the same fundamental methods. The difference is only in that this combines and streamlines the process with automated algorithms. When compared to unpublished data no difference in FRET efficiencies were detect as was expected. It may be necessary to use a more robust method to test the performance of this analysis relative to other methods in the future.

* Eccentricity, *e*, is a measure of how circular or linear an object is. It is defined as, $e = \sqrt{1 - \frac{b^2}{a^2}}$ where *a* is the length along the major axis and *b* is the length along the minor axis.

Data Handling and Graphical User Interface

Overview

A major issue with pervious ways of analyzing smFRET and co-localizations studies was that the process was broken up into many different scripts. Data often had to be formatted carefully before it could be run through each script. Many sections of the analysis had to be done by hand slowing down the analysis and easily biasing the data. Thus, it was critical that the methods described in the previous section be easily accessible for users. To meet this requirement a GUI was designed using the principles of human centric design with much attention paid to limiting the user from making technical errors[15]. The interface always provides users with automatic options that can often go without adjustment. The intent is to ensure that the user does not unknowingly bias their data and will only need to input to the analysis assumptions obvious to the user based on the nature of the experiment.

A full walk through of the layout and operation of the interface is provided in Appendix A.

Features

The user is always presented with the fewest options necessary such that user can easily navigate the interface. Though often many options are available to the user they are provided step by step such that the user is only given more options after having completed the previous set of options. This gives the user the options to always adjust previous options while limiting the amount of presented controls.

When using the interface, the current session can be saved. This session save file stores options specific to that analysis, detected features, and results from the analyses. However, it does not store the raw data used in the analysis so as to not waste space. Instead the locations of the raw data are stored. The interface can load in previously saved session files and will return to where the user was in the analysis with all their adjustments as they were. This saving feature has several uses. First, it allows users to revisit previously analyzed data to confirm, inspect or adjust the analysis. This is important as pervious methods often did not save the setting setup by the user and thus it was often not possible to exactly recreate the data. Second, this makes it easy for users to batch process several similar videos that require the same or similar settings. The user can simply analyze the data of one video, save the session, load in a new video and save the new

session with the new video analyzed the same as with the first video. Third, it allows users to quickly try several sets of analysis and compare the results. This is important in experiments where several phenomena are expected but will be detected using significantly different parameters. Finally, this feature is valuable because it allows researchers to easily start and stop an analysis throughout the work day. Once a save location has been set the program will auto-save the session every 15 minuets and before quitting.

All of the algorithms that take considerable amounts of time to compute are available to run in parallel when possible. This is limited in some cases when the raw data is required such as for the generation of the image of the kymographs. In these instances, only the portion of the scripts that specifically requires the raw data will be run linearly to prevent a large broadcast overhead from the raw data.

References

1. Sekar, R. B. & Periasamy, A. Fluorescence resonance energy transfer (FRET) microscopy imaging of live cell protein localizations. *J. Cell Biol.* **160,** 629–33 (2003).
2. Roy, R., Hohng, S. & Ha, T. A practical guide to single-molecule FRET. *Nat.Methods* **5,** 507–516 (2008).
3. Blanco, M. & Walter, N. G. Analysis of complex single-molecule FRET time trajectories. *Methods Enzymol.* **472,** 153–78 (2010).
4. MATLAB. (2016).
5. Hegadi, R. S. A Survey on Traditional and Graph Theoretical Techniques for Image Segmentation. *Int. J. Comput. Appl.* 975–8887 (2014).
6. Chenouard, N. *et al.* Objective comparison of particle tracking methods. *Nat. Methods* **11,** 281–289 (2014).
7. Mortensen, K. I., Churchman, L. S., Spudich, J. A. & Flyvbjerg, H. Optimized localization analysis for single-molecule tracking and super-resolution microscopy. *Nat. Methods* **7,** 377–381 (2010).
8. Crocker, J. C. & Grier, D. G. Methods of Digital Video Microscopy for Colloidal Studies. (1996).
9. Anthony, S. M. & Granick, S. Image analysis with rapid and accurate two-dimensional Gaussian fitting. *Langmuir* **25,** 8152–8160 (2009).
10. Vass, G. & Perlaki, T. Applying and removing lens distortion in post production. *Proc. 2nd Hungarian Conf. Comput. Graph. Geom.* 9–16 (2003).
11. Löhndorf, M. & Melenk, J. M. On Thin Plate Spline Interpolation. *Lect. Notes Comput. Sci. Eng.* **119,** 451–466 (2017).
12. Liu, J. *et al.* Cascading MutS and MutL sliding clamps control DNA diffusion to activate mismatch repair. *Nature* **539,** 583–587 (2016).
13. Bronson, J. E., Fei, J., Hofman, J. M., Gonzalez, R. L. & Wiggins, C. H. Learning rates and states from biophysical time series: A Bayesian approach to model selection and single-molecule FRET data. *Biophys. J.* **97,** 3196–3205 (2009).
14. Senavirathne, G., Mahto, S. K., Hanne, J., O'Brian, D. & Fishel, R. Dynamic unwrapping of nucleosomes by HsRAD51 that includes sliding and rotational motion of histone octamers. *Nucleic Acids Res.* **45,** 685–698 (2017).
15. Norman, D. A. *The Design of Everyday Things*. (Basic Books, 2002).

Appendix A

The program starts by asking the user whether they would like to load a previous session to make a new session. If the user chooses to load an old session, then the user is brought to where they were in the interface with the data they were exploring loaded with their setting. If the user chooses to start a new session, then they are asked to select a directory to save the session save file to and to export data to. The user is then presented with the home interface as shown in Figure 5. On the top of all the interfaces is a toolbar with buttons, from left to right, that load session save files, saves the current session, zoom-in any current image, zoom-out the current image, and to view the value at a position on a plot. In all interfaces the settings are located on the left and the data is presented on the right.

The user can then choose to enter four separate interfaces. The first interface available is the video settings interface. Initially the user is limited to two options: to select a video to analyze or to return to the home interface as shown in Figure 6. Once the user selects a video the interface adds the options to adjust the brightness of the video, cut frames from the video, crop the frame of the video, invert the video grayscale and to play the video as shown in Figure 7. From here the user can only adjust these settings or return to the home interface. After selecting the video, it will be displayed on the home interface as shown in Figure 8.

The second interface available to the user is the channel registration interface. Initially the user is limited to choosing a mapping video to load in, to load in the mapping from a previously mapped session which uses the same mapping, or to return to the home interface as shown in Figure 9. Once a video is selected the user is presented with options to adjust the brightness and to select the section of the video containing each channel as shown in Figure 10 and Figure 11. Once the user has finished selecting the channels the user is presented with options to detect the beads as shown in Figure 12. The user can then click “Register Channels” to complete the analysis as shown in Figure 13. When the user returns to the home interface they can now view the register video they imported in the video settings interface as shown in Figure 14.

The third interface available to the user is the DNA selection interface. Initially the user is limited to choosing a source to detect DNA from as shown in Figure 15. After a source is selected the user is presented one by one with the controls to adjust the DNA detection and selection. For each option an initial automatic value is used but the user is provided with some

options to adjust based on the nature of the experiment as show in Figure 16. The user can the proceed to generate kymographs. The kymograph interface provides users with the options of viewing each kymograph generated and to export these kymographs as shown in Figure 17. They can also make traces over the kymographs to retrieve dwell time and diffusion rates.

The forth interface available to the user is the FRET selection interface. Initially the user is limited to choosing a source to detect FRET from as shown in Figure 18. After a source is selected the user is presented one by one with the control to adjust the FRET detection and selection. For each option an initial automatic value is used but the user is provided with some options to adjust based on the nature of the experiment as show in Figure 19. The user can then proceed to generate traces. In the trace interface the user is presented with a variety of export options as shown in **Figure 20**.

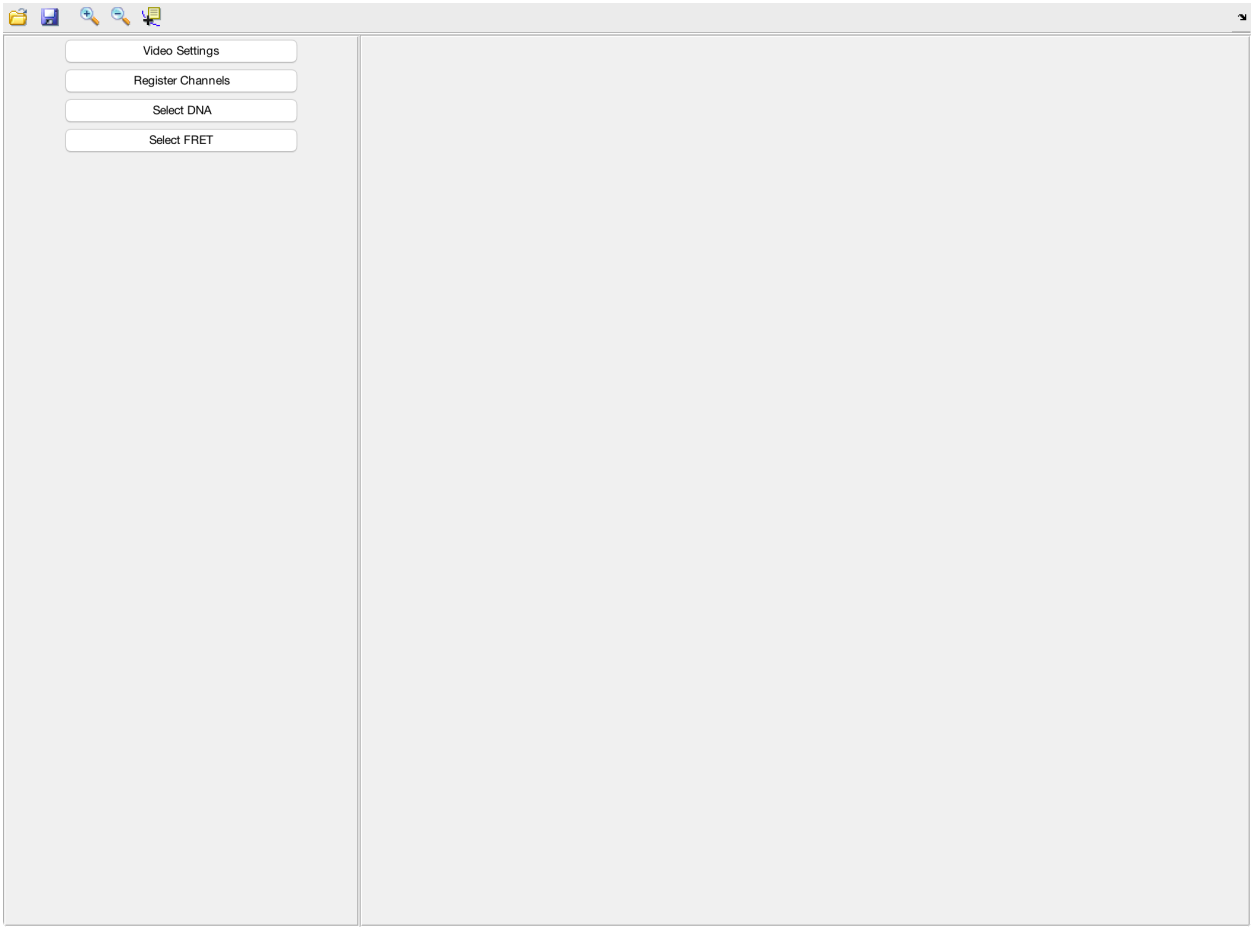

Figure 5 The home interface with no data loaded.

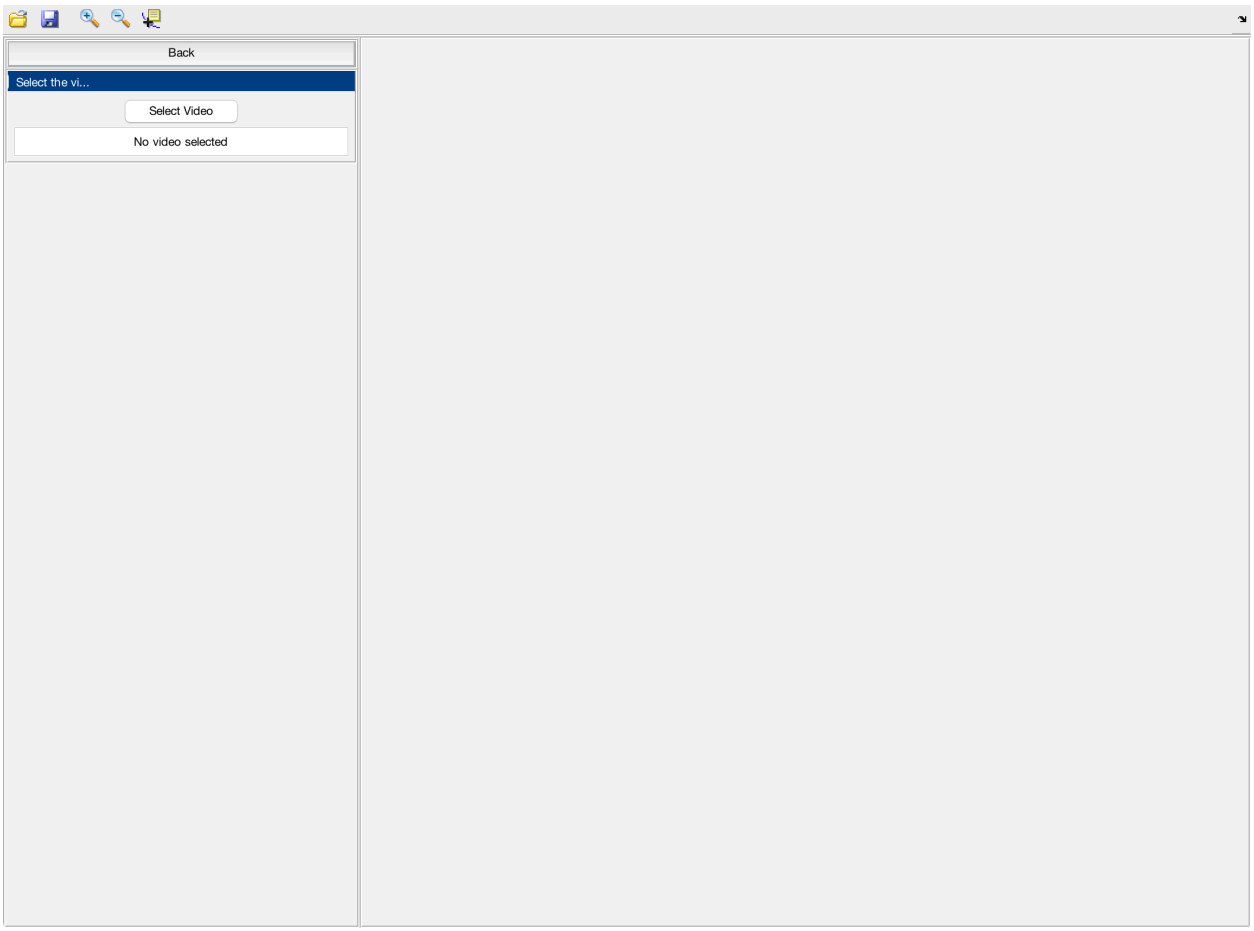

Figure 6 The video settings interface before any data is selected.

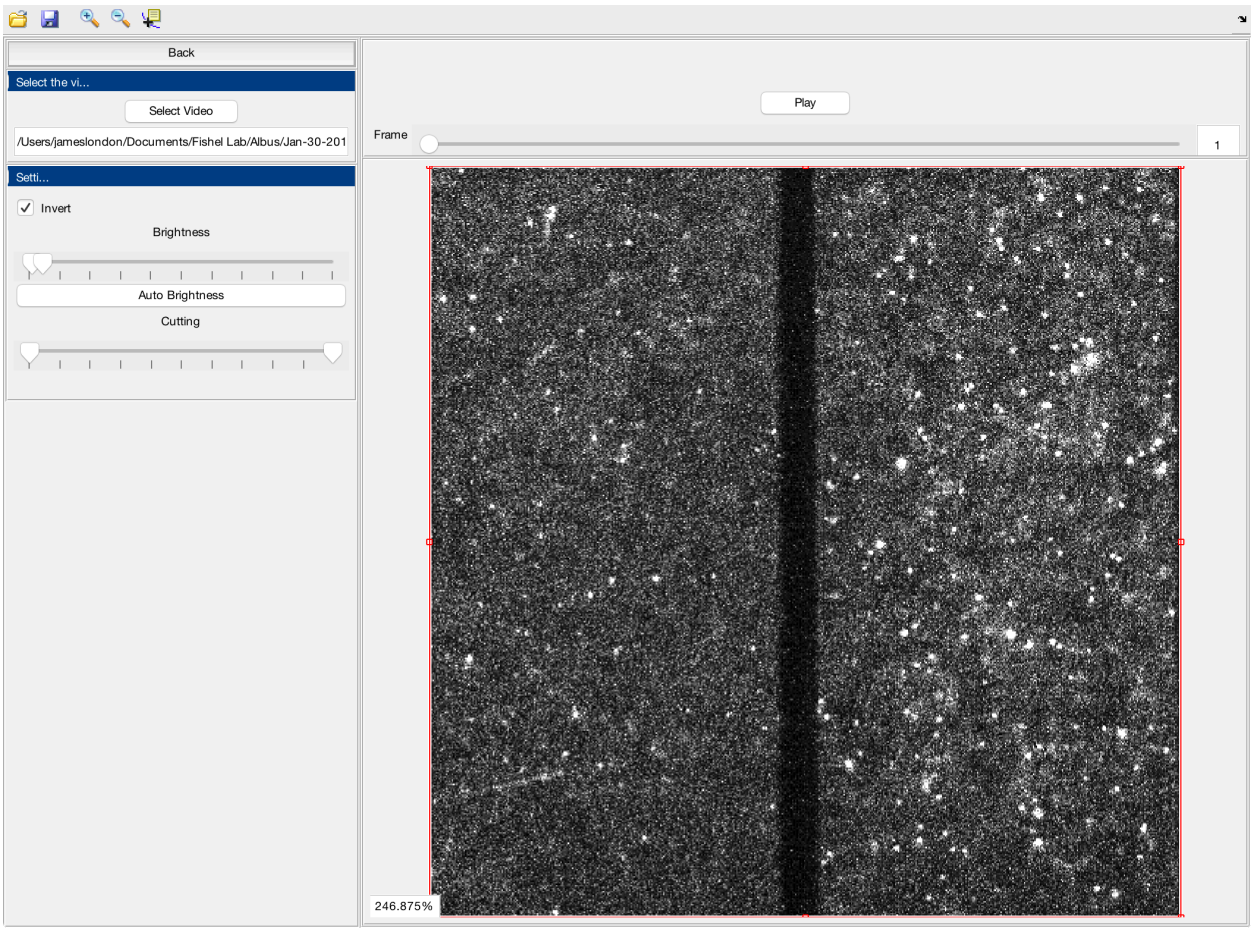

Figure 7 The video settings interface after a video is selected. On the left panel a new set of options is available to adjust the brightness of the video, to frames from the video, and to invert the colors of the video. On the top right panel, a play/pause button and a current frame slider are available to view the video. In the main panel in the bottom right the video is displayed with a magnification box in the bottom left and an adjustable border to crop the frame of the video. From here the user can only adjust the listed settings and return to the home interface.

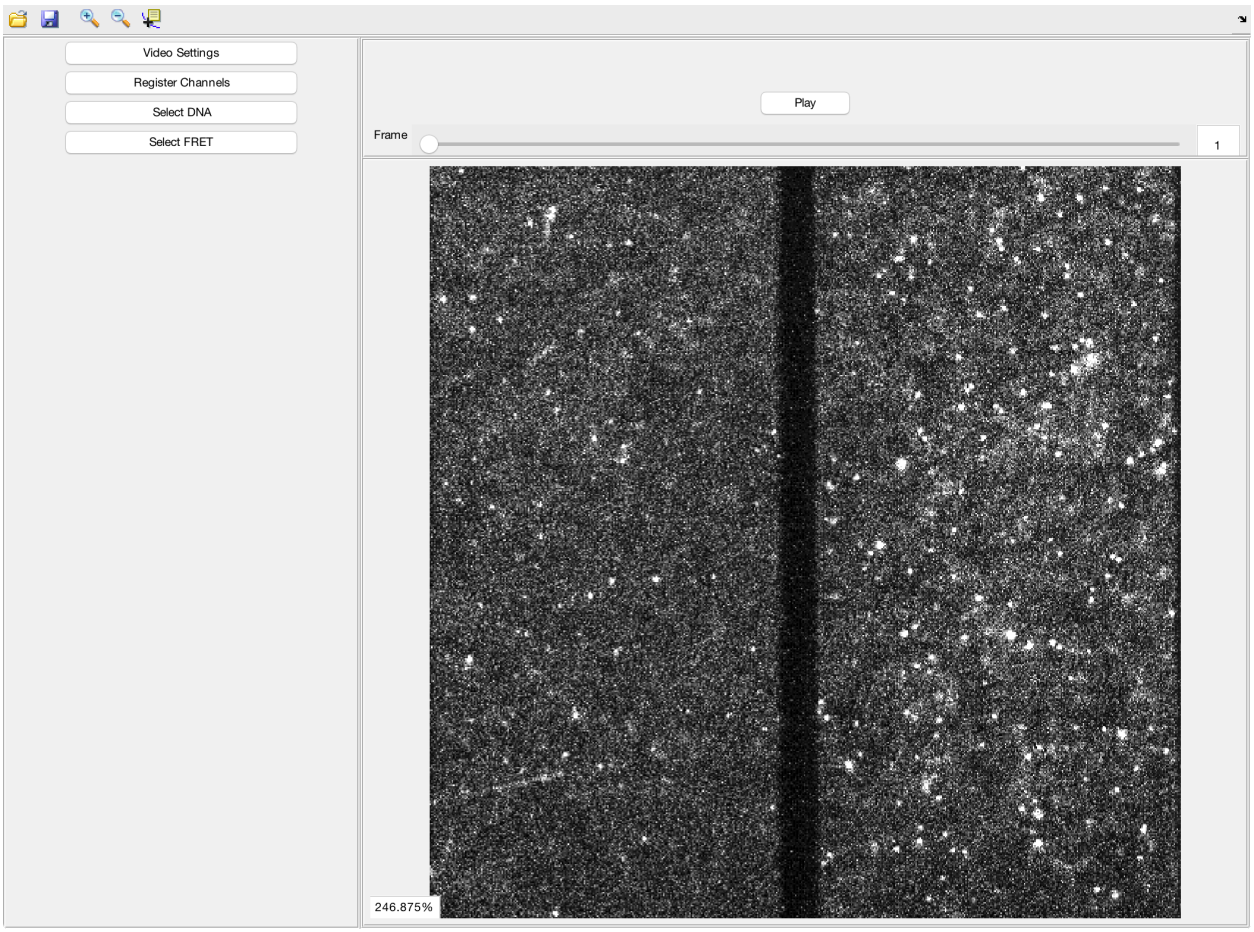

Figure 8 The home interface with data loaded after a video has been loaded but the channels have not been registered.

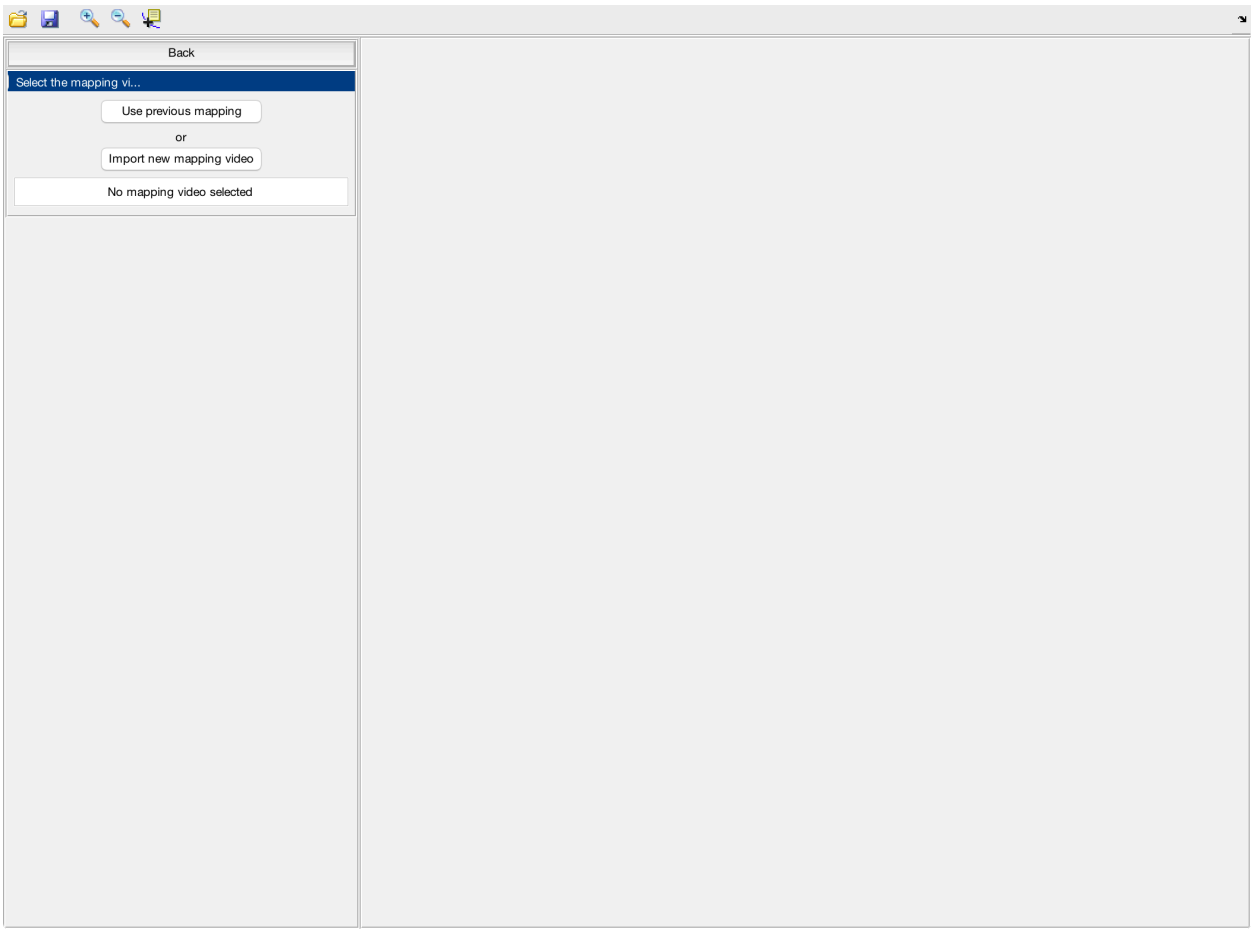

Figure 9 The channel registration interface with no mapping video loaded.

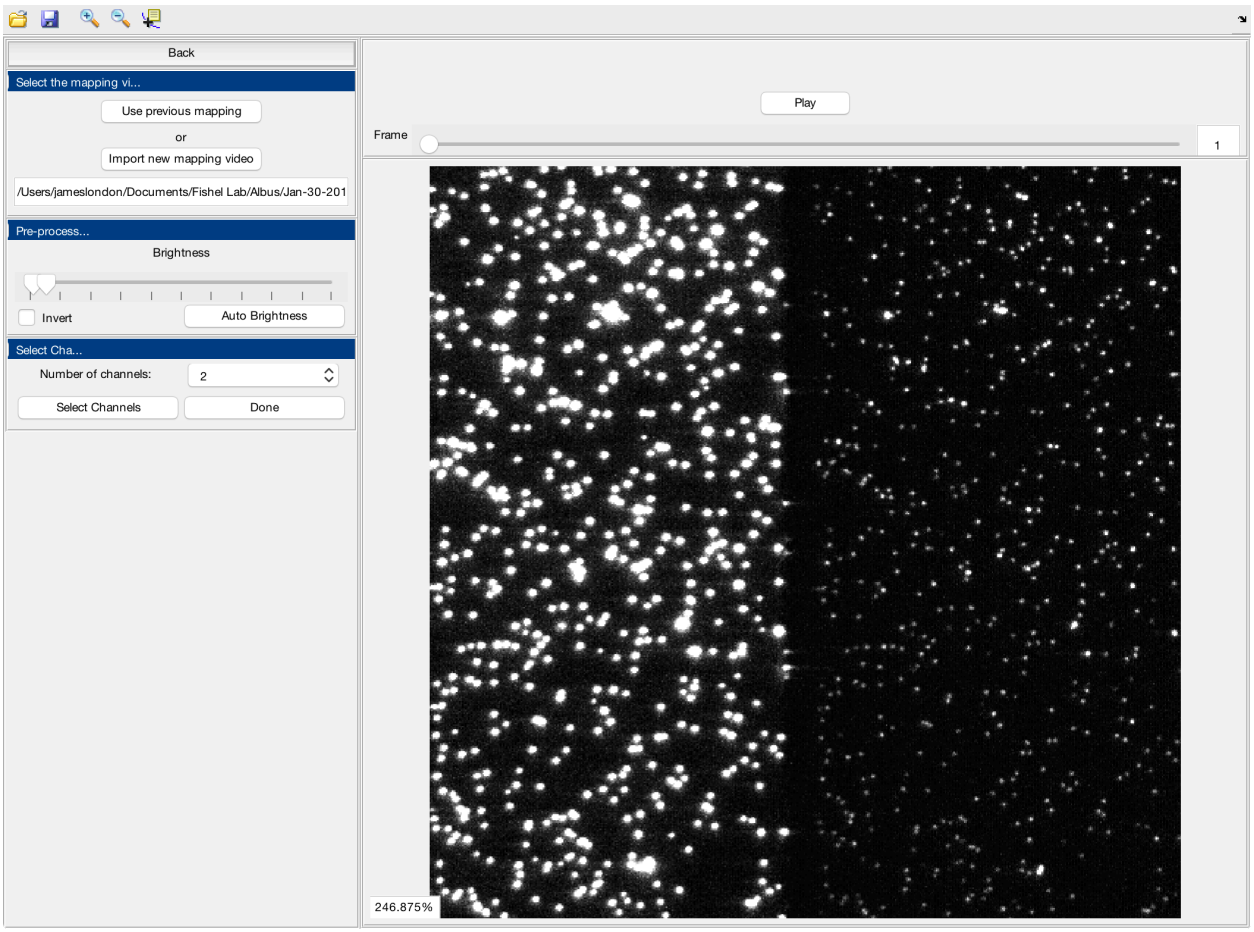

Figure 10 The channel registration interface after a video is loaded in.

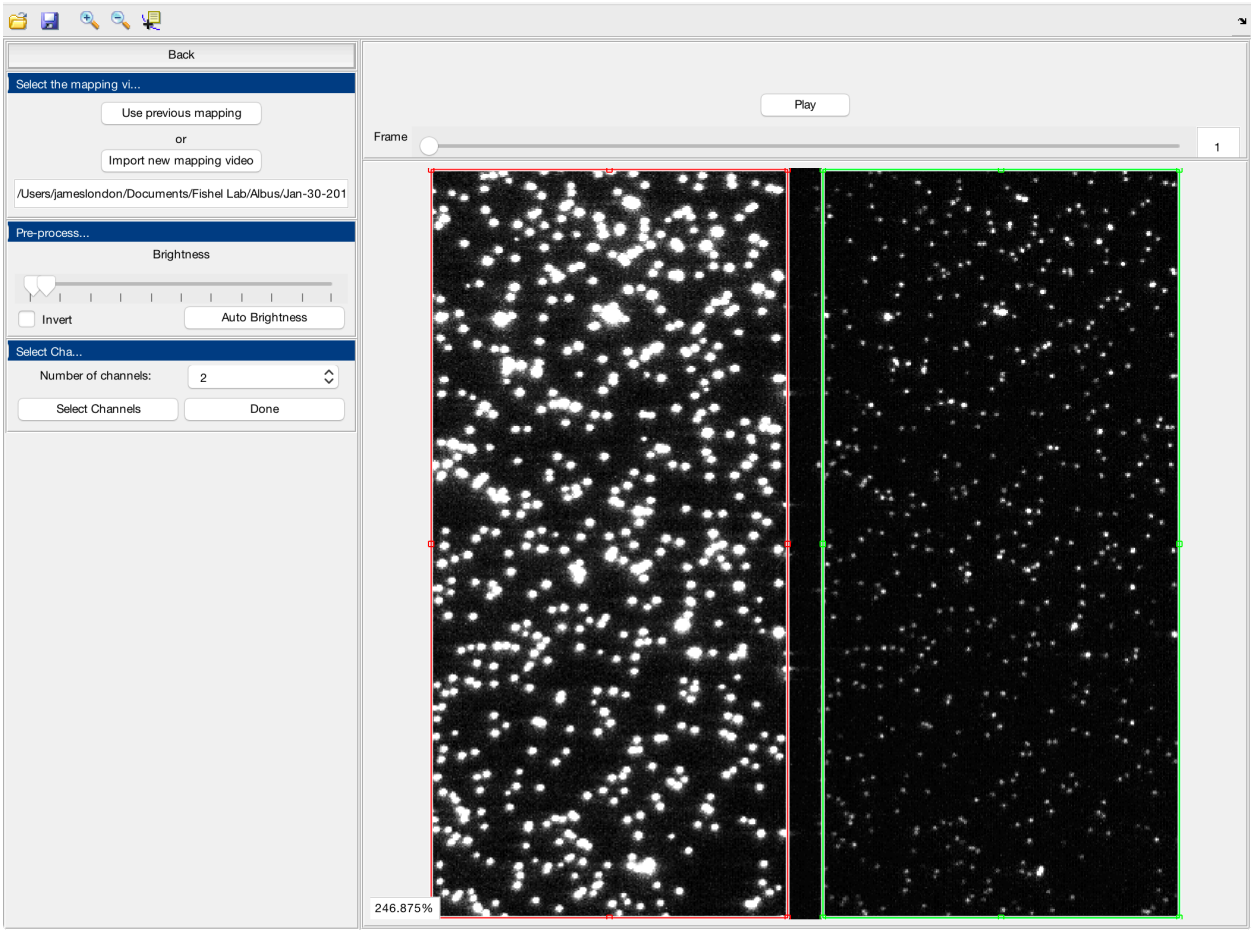

Figure 11 The channel registration interface after a video has been loaded and while the channels are being selected by the user.

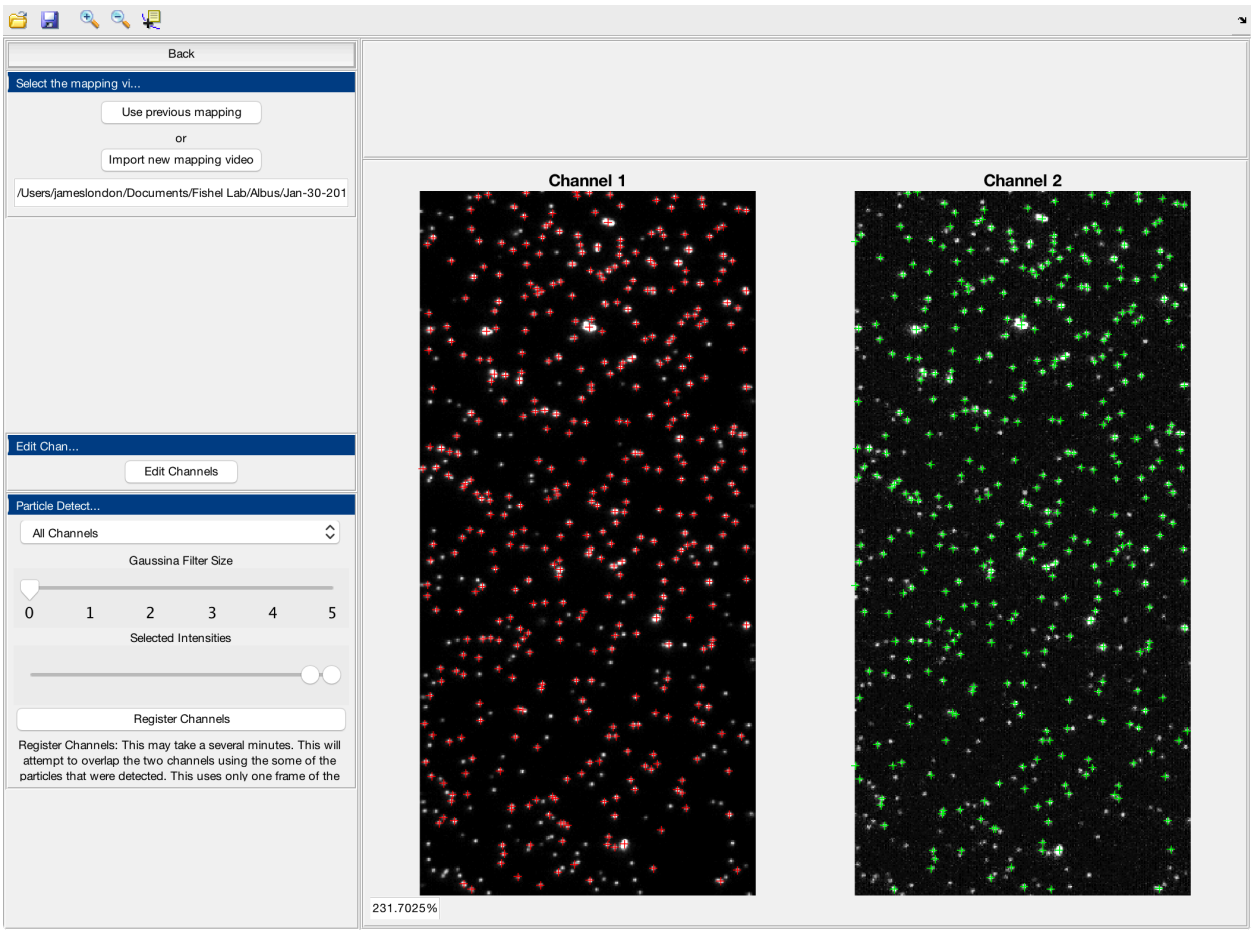

Figure 12 The channel registration interface after a loaded mapping video's channels have been separated while the mapping beads are selected.

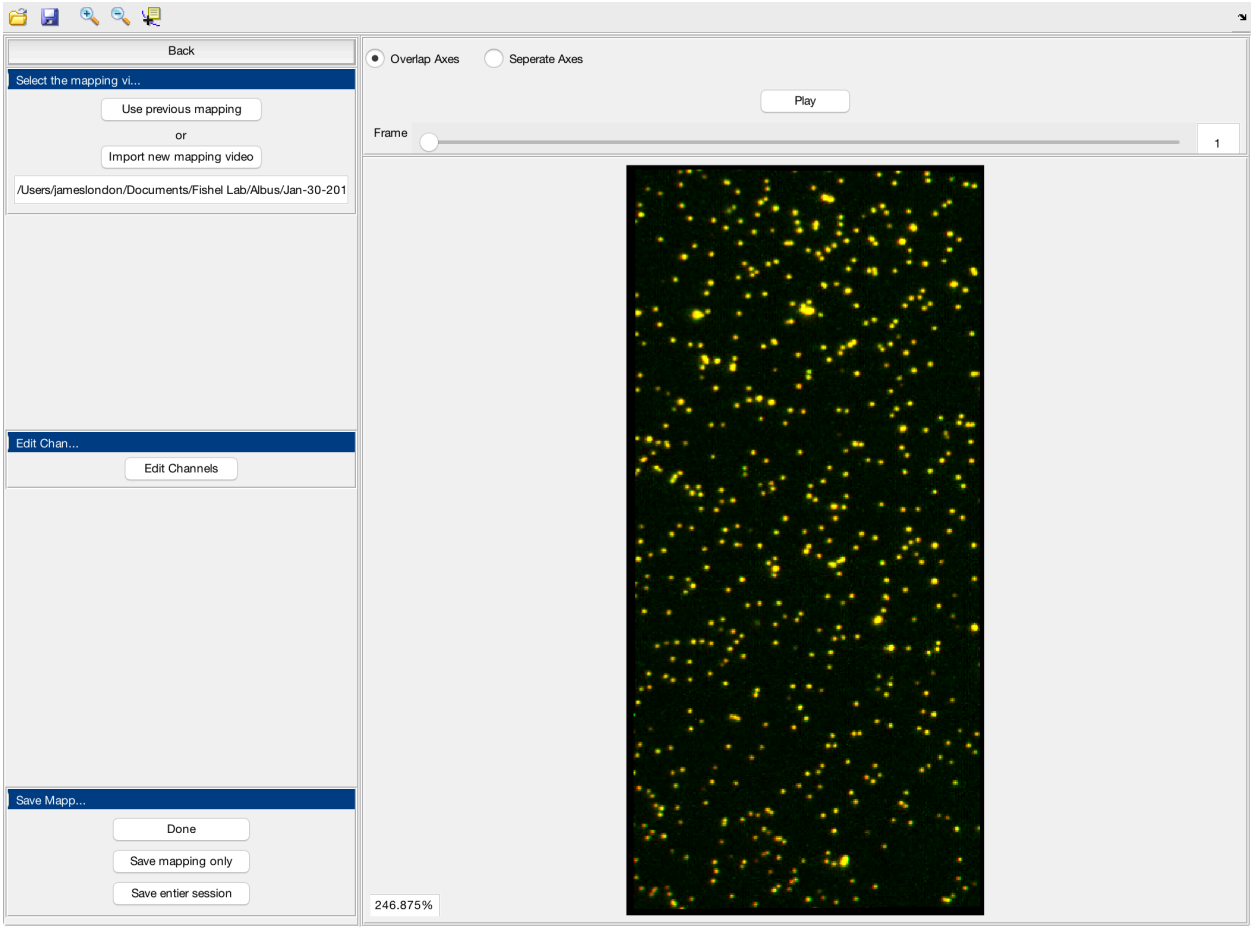

Figure 13 The channel registration interface after the channels have been registered.

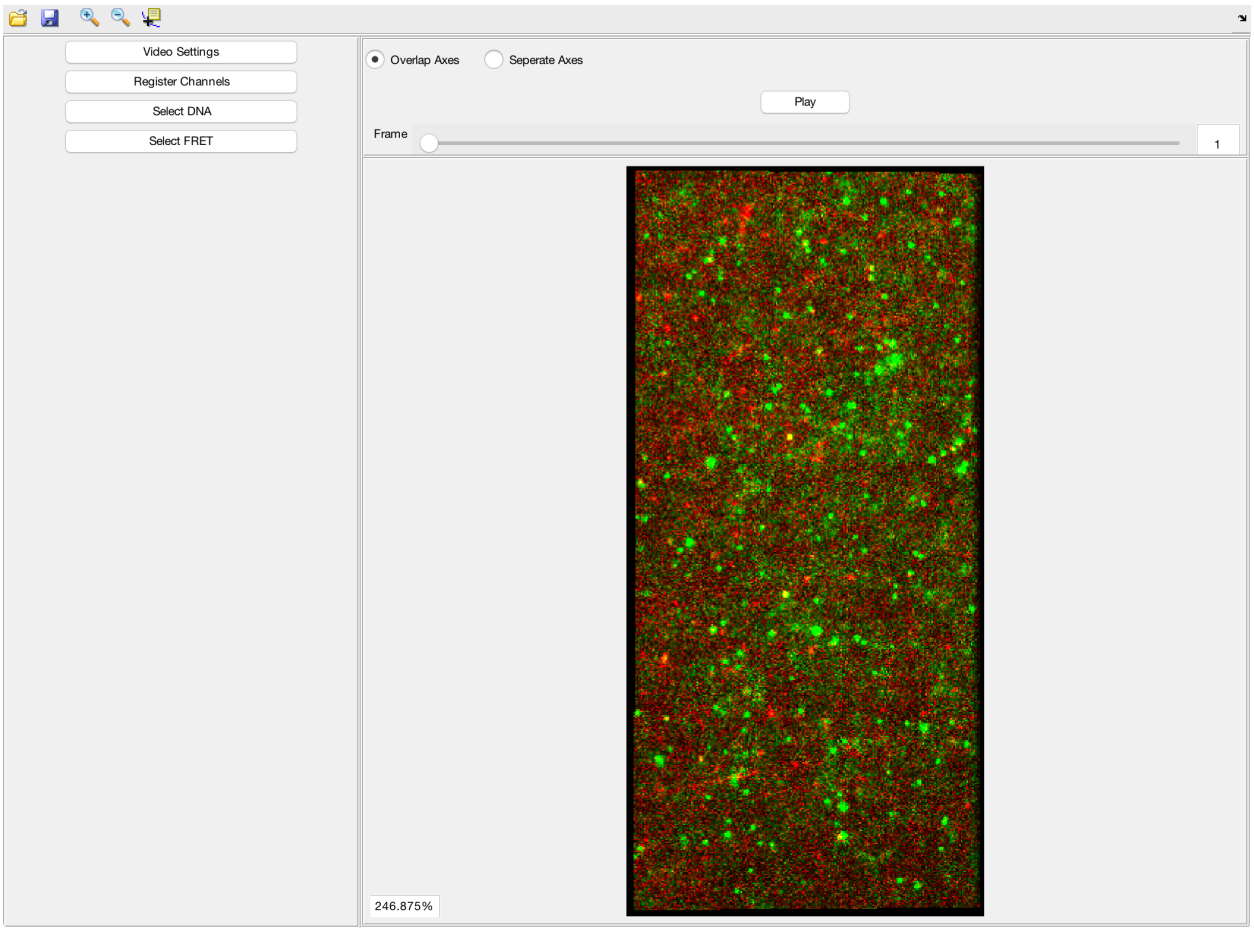

Figure 14 The home interface after a video has been selected and the channels registered.

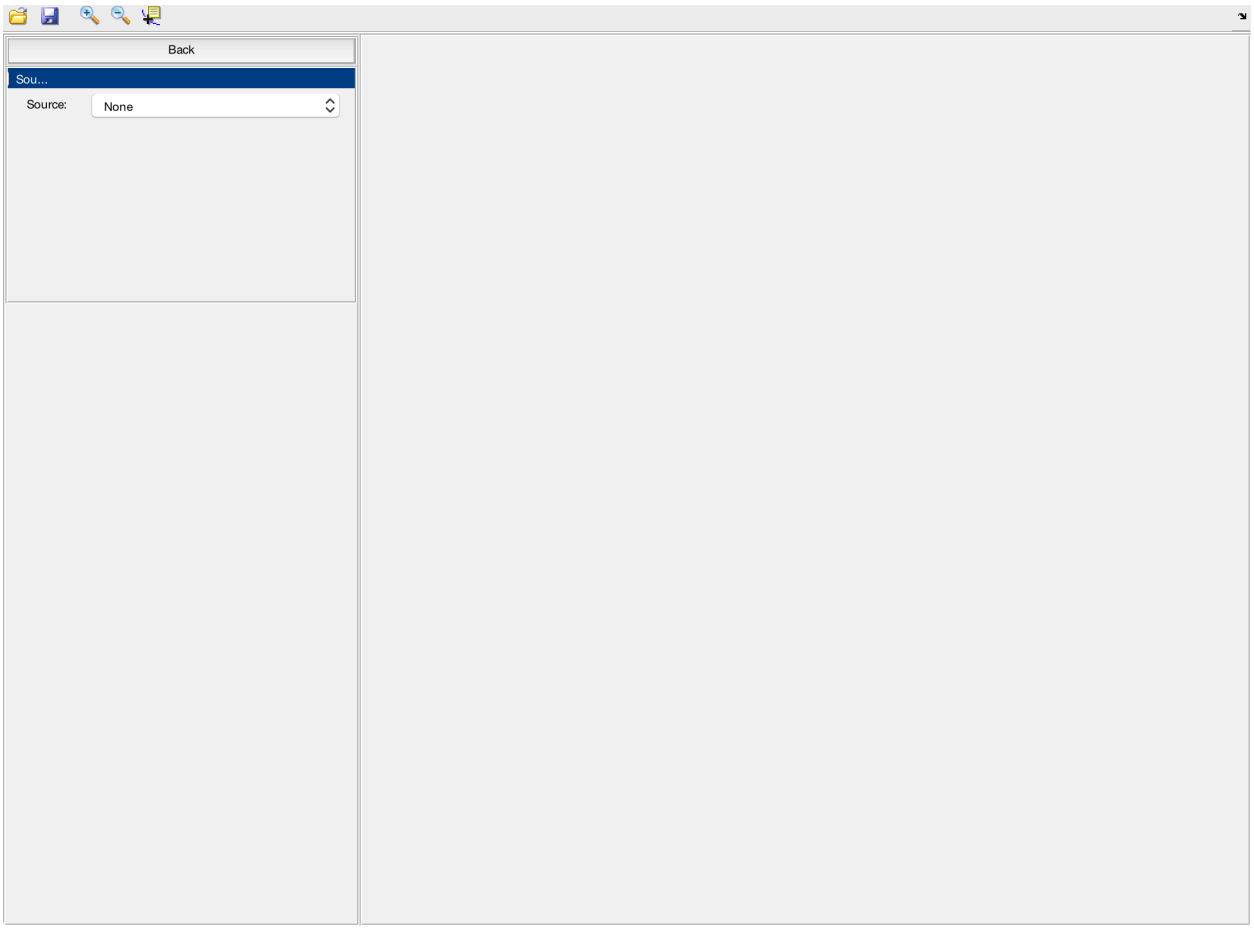

Figure 15 The DNA selection interface before a source for detecting the DNA is selected.

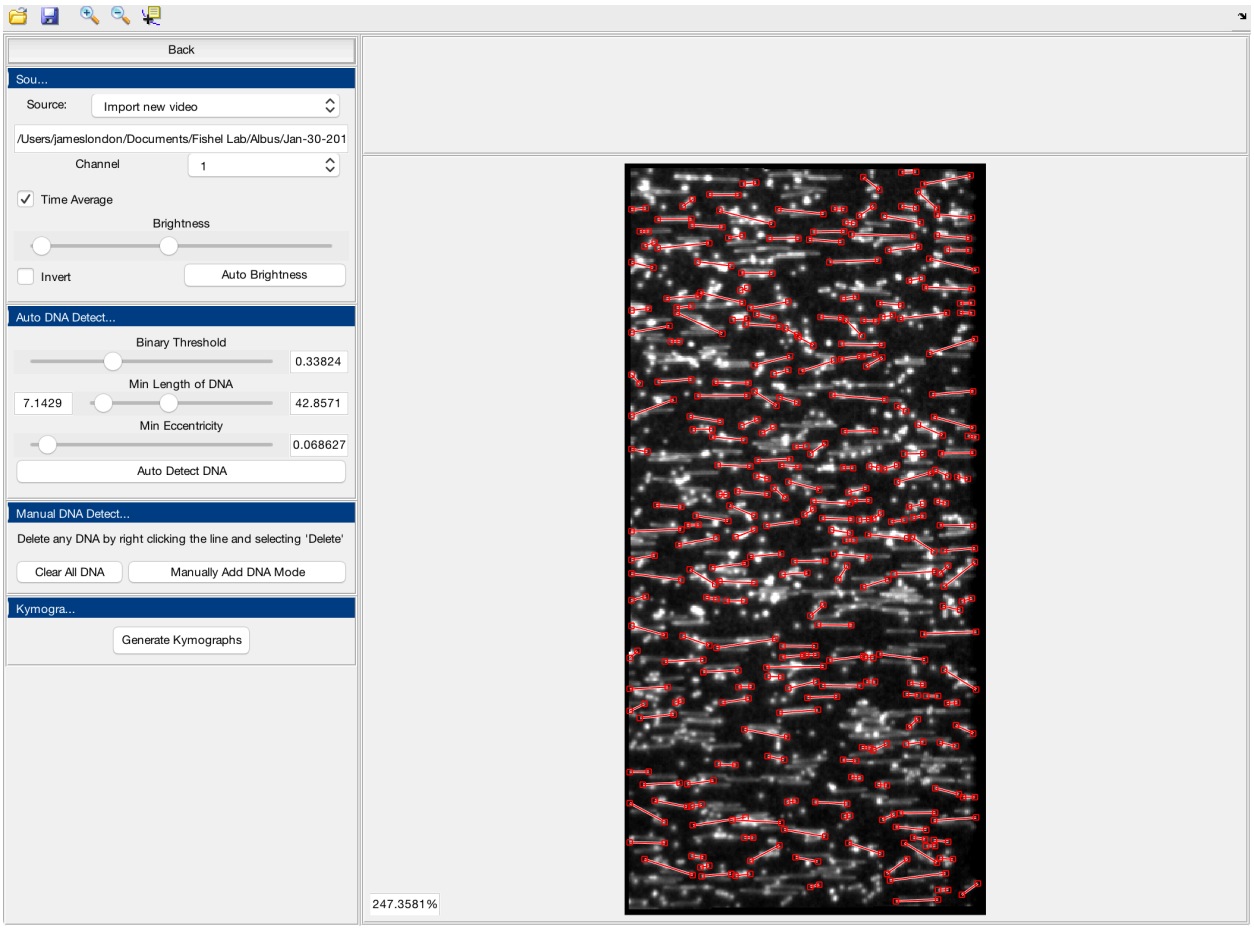

Figure 16 The DNA selection interface with a stained video imported and DNA automatically detected.

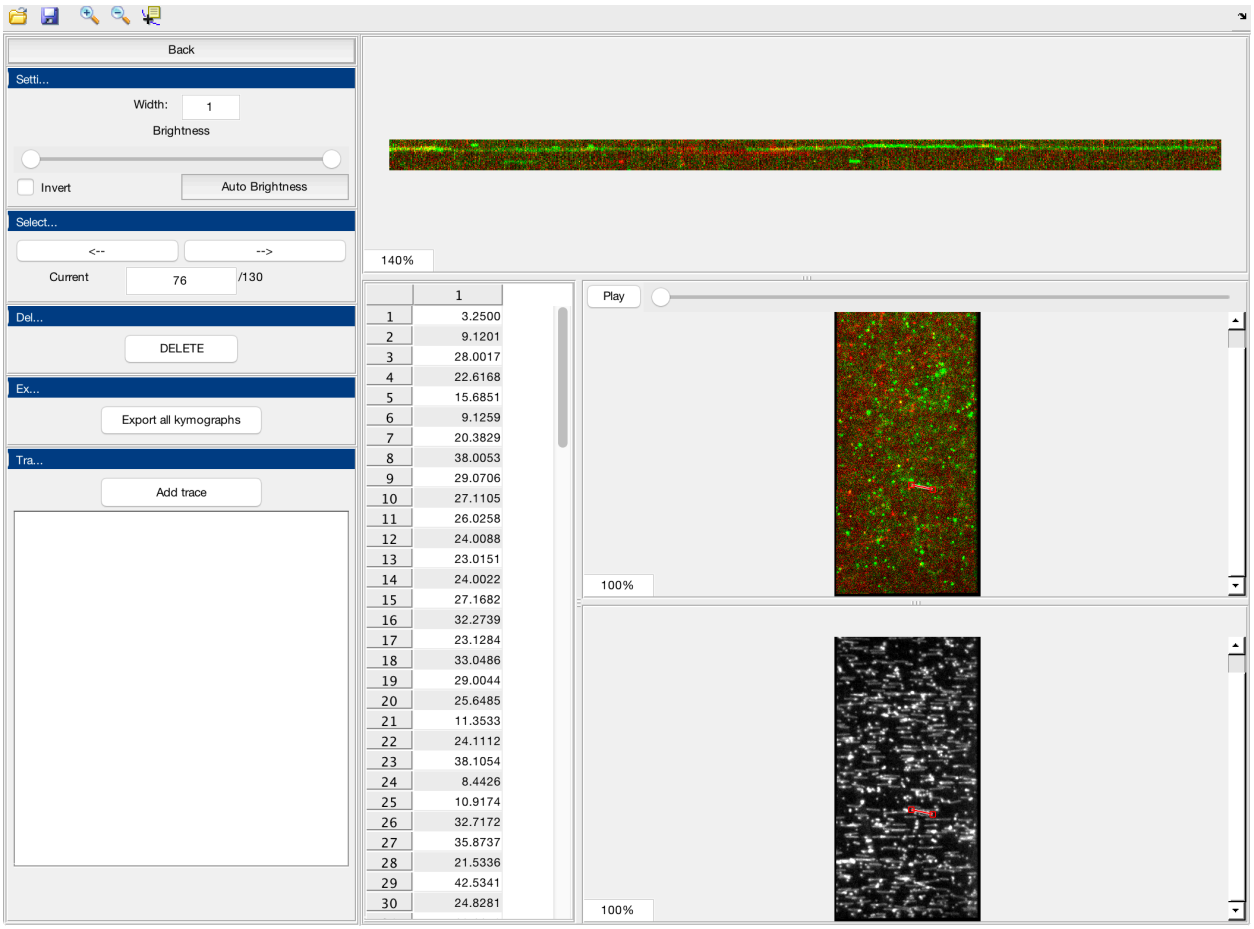

Figure 17 The kymograph interface with no traces selected.

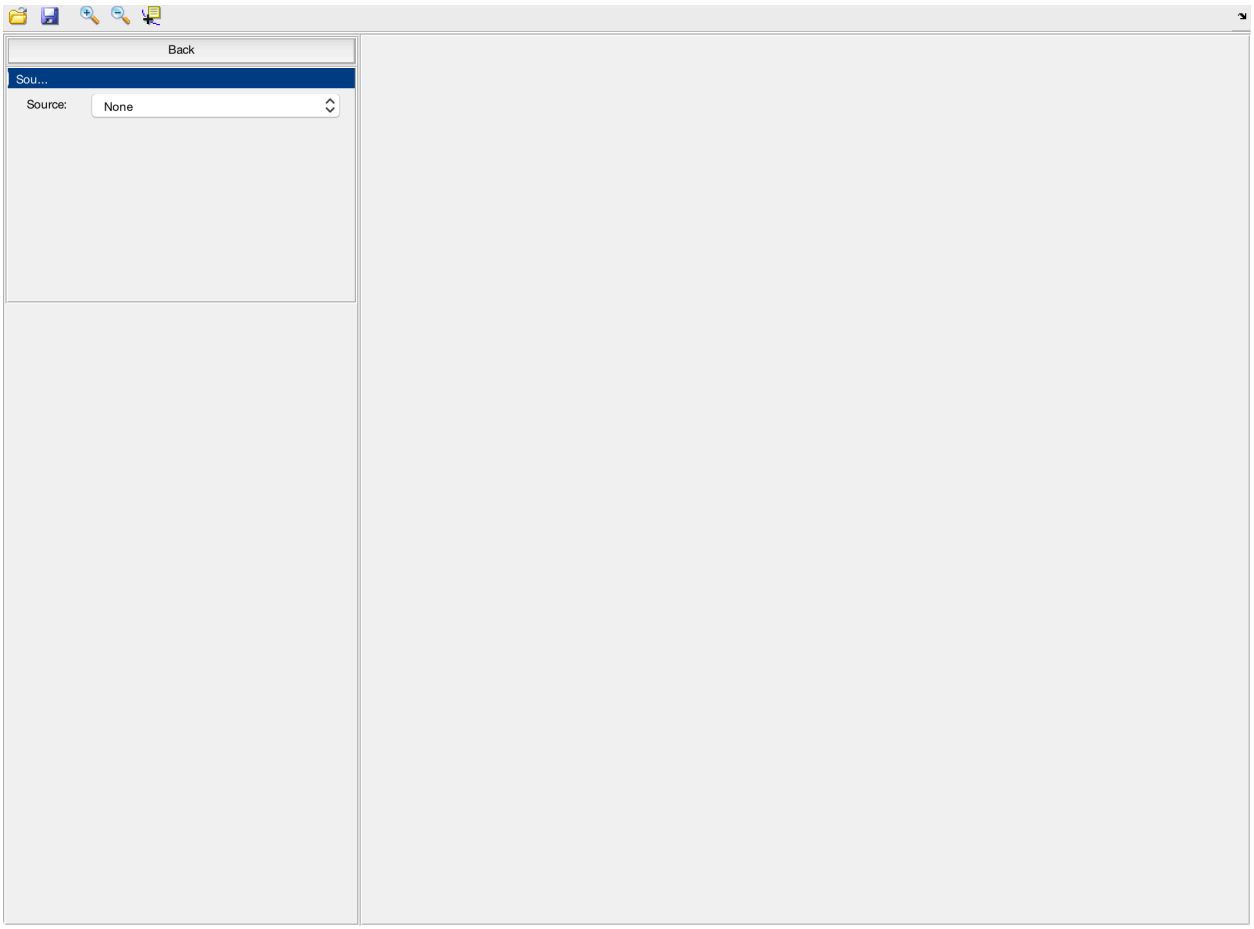

Figure 18 The select FRET interface before a source of the FRET locations is selected.

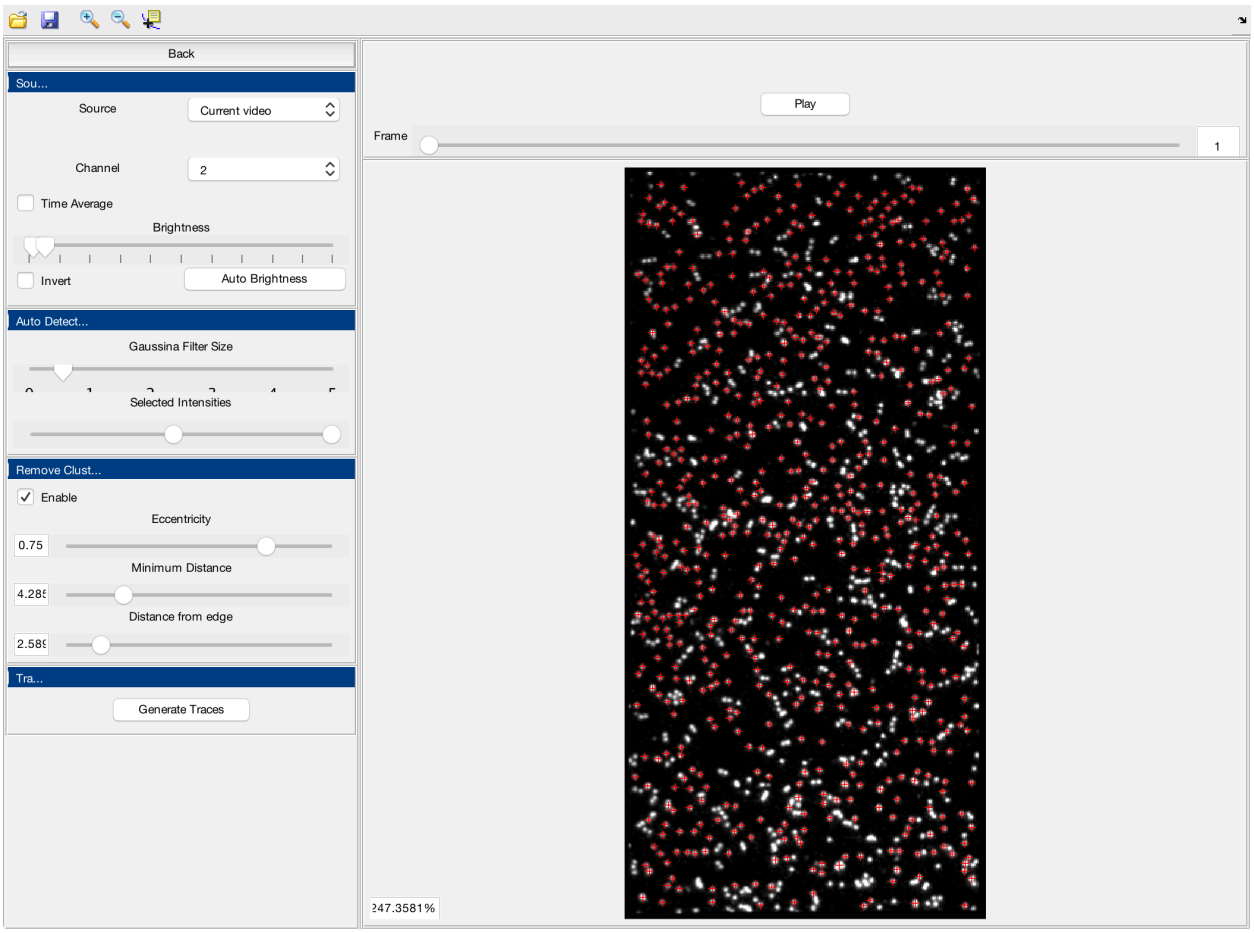

Figure 19 The select FRET interface after the first frame of the current video is selected as the sources with automatic selection of FRET locations overlaid.

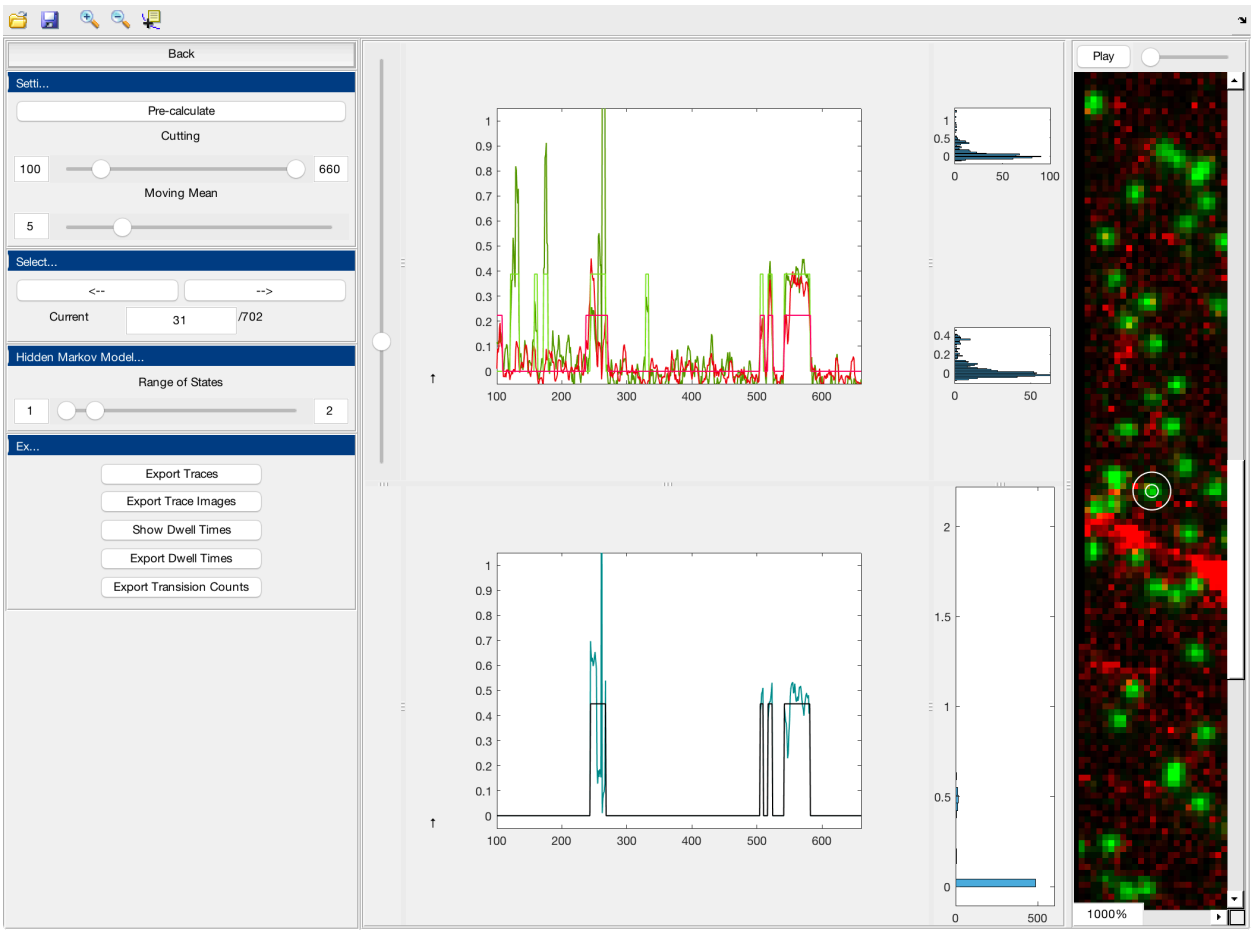

Figure 20 The trace interface showing the acceptor, donor and FRET efficiency traces.